Iqbal, Ahmed Ali and Mahmood, Waqar and Ahmed, Ejaz (2005) ***TCP-DR an effective protocol for infrastructure based wireless networks.*** In: Proceedings of the 2005 International Conference on Wireless Networks, ICWN '05, 27 - 30June 2005, Las Vegas, Navada.

TCP-DR an Effective Protocol for Infrastructure based Wireless Networks

Ahmad Ali Iqbal, Waqar Mahmood, Ejaz Ahmad
Communication System Engineering Department, NIIT
National University of Science and Technology (NUST), Rawalpindi, Pakistan

Abstract - TCP is a dominant protocol for consistent communication over the internet. It provides flow, congestion and error control mechanisms while using wired reliable networks. Its congestion control mechanism is not suitable for wireless links where data corruption and its lost rate are higher. The physical links are transparent from TCP that takes packet losses due to congestion only and initiates congestion handling mechanisms by reducing transmission speed. This results in wasting already limited available bandwidth on the wireless links. Therefore, there is no use to carry out research on increasing bandwidth of the wireless links until the available bandwidth is not optimally utilized. This paper proposed a hybrid scheme called TCP Detection and Recovery (TCP-DR) to distinguish congestion, corruption and mobility related losses and then instructs the data sending host to take appropriate action. Therefore, the link utilization is optimal while losses are either due to high bit error rate or mobility.

Keywords: Bit Error Rate (BER), congestion control, Congestion Window (cwnd), Handoff, TCP performance.

1.0 Introduction

TCP was developed for reliable communication in wired networks as end to end transport layer protocol. It provides congestion control, error control and flow control mechanisms for guaranteed delivery of data. Its congestion control mechanism is transparent from the physical link (either it is wireless or wired). TCP was proposed for consistent and reliable [2] wired medium. All packet losses were considered as due to congestion at intermediate nodes and congestion handling mechanisms like slow start and fast recovery, were initiated to reduce the transmission speed. TCP congestion control mechanism is still efficient and effective for wired networks but is inefficient for wireless links because they depict different characteristics than wired ones. Mobile hosts have different needs and limitations than desktop computers like their transmission media, its reliability of transmission, packet size limitations etc. Therefore, the throughput in a given interval of time is fairly lesser on the wireless links while using traditional TCP. That is why it significantly, affects the TCP performance when multiple packets from the advertised window are lost [12].

TCP uses two kinds of windows for reliable transmission, flow control and congestion control. (i) Advertised Window; its value is determined during negotiation of sender and receiver while connection establishment phases takes place (ii) Congestion Window (cwnd); its value is modified when TCP experiences congestion (it is determined by loss of packets or time out at the source host). TCP can transmit maximum number of unacknowledged packets either from advertised window or from cwnd what ever is minimum. The value of cwnd is directly linked to packet losses and is reduced each time, TCP experiences packet losses that may be determined by either through duplicate acknowledgements (normally three) or time out, without taking into account the reason of packet loss i.e. congestion, bit error rate or mobility. Due to frequent packet losses [5] in wireless networks, the value of cwnd remains very small.

TCP initiates congestion control mechanism by initiating slow start, congestion avoidance or fast recovery mechanisms depending on *cwnd*, *ssthresh* etc on detection of loss of packet and resets its retransmission timer [4]. Therefore, the reaction of TCP towards the congestion control results in an unnecessary reduction in the link's bandwidth utilization [2]. This phenomenon significantly degrades the overall performance of wireless networks in the form of lower throughput along with high delays.

Research work defined in this paper is a logical extension to the work described in the paper [1]. The rest of the paper is organized as follows: Performance factors are discussed in Section 2. Related work is presented in Section 3. In section 4, we have provided the comparative analysis of the protocols discussed in section 3. Our proposed architecture of TCP-DR is discussed in Section 4. Section 5 contains the future research work and conclusion.

2.0 Performance Factors

The main factors [1] that effects TCP performance in mobile wireless networks are as follows

a. High bit error rate [11][12]*:* wireless links are severely affected by the environmental changes. Therefore, they are susceptible to high bit error rate that causes the data or acknowledgement corruption or loss. The performance comparison at BER 10^{-5} and 10^{-6} using TCP Reno is shown in table 1 [9].

Table I. Effect of BER on performance of TCP

BER	10^{-5}	10^{-6}
Throughput (pkts/sec)	39.439	87.455
Success Probability	0.9892	0.999
Transfer time of 5000 pkts in sec	123.847	58.032

b. Disconnections [8][9]*:* Frequent disconnections become the cause of serial timeout. Serial timeout is assumed to be a strong form of congestion thus causes slow start and regular disconnection makes the *cwnd* to be very small. The disconnection can be due to handoff, bandwidth, connectivity and movement of mobile node outside the access point range.

c. Cell size [9]*:* Each cell has its own access point and the bandwidth available to each mobile node is dependent on the number of nodes in a cell. If cell size is larger, there is more chance to increase mobile nodes within a cell hence less bandwidth for each node. But making cell size small; increases the handoff due to frequent disconnection. This requires more state shift from one base station to the other. Therefore, the selection of an appropriate cell size is important.

d. Power consumption [5][8]*:* There is a limited power available to mobile node because they operate and dependent on the battery power. Therefore, few TCP enhancements cater for the minimum power consumption.

e. Connection: Compared to traditional wired network, wireless networks are much more convenient to setup. Most Wireless LANs support ad-hoc networking which allows quickly building up a network and using it without any infrastructure. On top of that, users are able to move around while still connected to the network.

f. Packet size variation [9]*:* The packets transmitted in a wireless link are normally smaller than the packets in wired networks. Thus, the packets coming from wired network for the wireless networks are segmented into smaller chunks at base station. So, that only a smaller data packet could be retransmitted, if it is corrupted or not received to the mobile node. Segmentation of a packet increases overhead due to the number of headers along with each smaller chunk instead of one header for each packet.

3.0 Related Work

Researchers have proposed solutions to improve the performance of TCP over mobile wireless networks in an attempt to provide seamless internetworking between the wired and wireless worlds. In this section, we compare these TCP enhanced protocols performance under the different parameters.

3.1 The Snoop Protocol aimed to achieve the goal of improving TCP performance without changing the existing TCP implementation in the fixed network [4]. It is a link layer protocol that introduced a module, called Snoop, at the base station that monitors every packet passing through in either direction. The Snoop module maintains a cache of TCP packets sent from the fixed host that have not yet been acknowledged by the mobile host. To implement the local timeout, the module has its own retransmission timer. The Snoop module retransmits the lost packet if it has it in the cache. Thus, the base station hides the packet loss from the fixed host and avoids an unnecessary congestion control mechanism initialization.

The improvements in snoop protocols are: (i) By adding selective retransmissions from the base station to the mobile host and the case when the packet losses are from the mobile host to the base station. (ii) In order to have the sender identify whether the loss happened on the wireless link or on the wired link due to congestion, the snoop module keeps track of the packets lost at the base station. The base station then generates negative acknowledgements for those packets back to the mobile. This is very useful when more than one packet is lost in the same window.

3.2 EBSN takes the effect of local error recovery and explicit feedback by the base station as a primary factor. EBSN is an end-to-end protocol [6]. The idea is that although local recovery by the base station using link layer protocols is found to improve performance, timeouts can still occur at the fixed host, causing redundant packet retransmissions. Their experiments showed that explicit feedback from the base station to the fixed host can completely eliminate the possibility of timeouts occurring at the fixed host, while the wireless link is in a bad state. When a wireless link is in a bad state, little data is able to reach the mobile, causing packets from the fixed host to be queued at the base station. The protocol also considered the effect of packet size variation and proves that selecting optimal packet size could significantly improve performance.

3.3 MTCP: is to protect the long connection over the wired network from the impact of the erratic behavior of the short connection over the wireless link and also recover quickly from errors over the wireless link [13]. MTCP has introduced a session layer protocol called MHP (Mobile Host Protocol), at the base station and the mobile host because this layer compensates for the unreliability and unpredictability of the wireless link using its knowledge about host migration and wireless links characteristics. Two implementations for the session layer are proposed. One uses TCP over the wireless link, and the second uses a selective repeat protocol (SRP) over the wireless link. SRP is designed to recover quickly from high and bursty packet losses. The receiver returns a selective ACK (SACK) when an out of sequence packet is received specifying the missing packet. Then, the sender in turn retransmits the missing packet. SRP also can recover more than one packet in one round trip time.

3.4 E-TCP: Here, it is suggested that an acknowledgement scheme is to be used. There is an agent [12] in the intermediate node located at the edge of wired and wireless link whose function is to distinguish between the two losses (i.e. congestion or corruption loss). The agent explicitly acknowledges each packet transiting the base station. This new acknowledgment packet uses two extra bits Originating Bit (Ob) and Destination Bit (Db) from the reserved portion of TCP header to inform the sending host about the reason of loss. The sending host then takes an appropriate action by either shifting to slow start or by initiating fast recovery mechanism [5].

The protocol requires slide modification at sending host in TCP so that it becomes capable of handling the loss of packet due to different reasons. If both sender and receiver are FH then normal TCP recovery mechanism is adopted. Otherwise working of TCP is shifted to E-TCP for recovery of lost packets.

3.5 I-TCP splits a TCP connection between a fixed and mobile host into two separate connections at the base station [3]. One TCP connection is between the fixed host and the base station. Other connection is between the base station and the mobile host. First connection is a normal TCP connection. But the connection between base station and wireless node is considered to be one hop away. Therefore, there is no need to use TCP on this link. Rather, a more optimized wireless link-specific protocol tuned for better performance can be used. Most important advantage is Mobile Host implements very simple transport layer protocol for the communication with Mobile Support Router (MSR) by transferring its processing overhead to MSR. One of the problem arises in this approach is that splitting could not provide end-to-end connectivity as connection is physically divided into two separate connections. This increases complexity and software overhead by four times as compared to normal TCP. I-TCP violates the concept of end-to-end TCP connectivity by physically dividing the connection into two sub connections. I-TCP buffer overflows in Mobile Support Router (MSR), if the sender transmits the data larger than the size of buffer of MSR.

3.6 M-TCP provides the mechanism to increase performance by providing additional functionality to handle when there are periodic disconnections [8] that are normally due to the mobility of mobile hosts. M-TCP divides the geographical region into cells where each cell may contain one base station. M-TCP suggests three level hierarchies to provide connections for a long run. Lowest level consists of mobile nodes in which mobile hosts (MH) communicate with their mobile support stations (MSS). At the middle level, Supervisor Host (SH) is usually connected through wired networks and controls multiple MSS via physical connections. It has the responsibility to deal with routing and other functions for the mobile users. At the top level of the hierarchy, high speed wired network exists in which Fixed Hosts (FH) are connected to this high speed network.

The sender is forced to the persistent state by freezing *cwnd* and *RTO* in the case of disconnection. After receiving notification message from M-TCP, the sender resumes it states as it was preserved and the disconnection has no effect on the performance even if there are multiple disconnections during hand-offs of MH.

4.0 Comparative analysis of TCP performance enhancement protocols

The advantage of the EBSN [6] scheme is that it eliminates timeouts at the fixed host during local recovery. This will provide great performance improvement, especially in wireless links with high bit error rates. The major disadvantage is that handoff due to mobility is not considered.

MTCP is suitable for one or two hop away base station [13] [3] and therefore not scaleable. E-TCP is also limited for the communication up to two base stations. Advantage of E-TCP over the other proposed schemes is that other schemes only provide the solution for the communication between a Mobile Host and a Fixed Host whereas E-TCP provides the solution for all possible combinations of communication that may be between mobile and fixed hosts.

The main drawback of snoop split connection approach is that the semantics of TCP acknowledgments are violated [4]. The snoop protocol maintains the end-to-end semantics of the TCP connection between the fixed and mobile hosts by not generating any artificial acknowledgments [9]. Handoffs in this approach require the transfer of a significant amount of state. For example, I-TCP handoff times vary between 265 and 1400 ms [7] depending on the amount of data in the socket buffers that need to be transferred from one base station to another. The snoop protocol performs handoffs based on multicast and typical completion times are between 10 and 25 ms. In M-TCP, handoffs are very low in number because of three level hierarchy [8].

M-TCP [8] has one more unique characteristic comparing to I-TCP and snoop protocols; it also caters for the power scarcity [3] because of low variable bandwidth and fewer disconnections. The basic difference between M-TCP and I-TCP is that M-TCP focuses on effect of motion and handoff while I-TCP is specific for the effect of small packet size and lossy links. I-TCP also uses TCP with a little modification to handle mobility but it is susceptible to power scarcity [3].

One more advantage of M-TCP over I-TCP is that I-TCP assumes one or two hops [3] between the mobile host and base station. Thus suitable for smaller RTT but M-TCP is suitable for large networks as a host moves from one cell to other without even need to transient states between the two base stations because state is maintained at super host that controls multiple cells and their base stations. In the case of snoop protocol, mobile host stay connected to the previous base station until it state is cached to the new base station therefore throughput goes down until new cache is built up.

5.0 TCP-DR (Detection & Recovery)

Most of the protocols suggested so for target either the bit error rate, the mobility of the node or the power scarcity. There is a need to develop a comprehensive protocol that caters for all or most of the factors simultaneously that affects TCP performance. The above analysis shows that M-TCP is best solution for handling mobility and disconnections while E-TCP is best for handling bit error rate. Our proposed scheme inherits the good characteristics from both the protocols to give best performance for the wireless networks when mobility is very frequent and high bit error rate.

5.1 TCP-DR at Base Station

TCP-DR requires modification to the TCP implementation at base station node where each segment passing through is stored and each connection segments are cached separately until an ACK from the data receiver for that segment is received. The purpose is to locally transmit the packets while they are corrupted or lost. Each packet is also time stamped by TCP-DR to locally timeout. This timestamp is lesser than RTT. The local RTT is calculated and modified time in the same way as TCP source calculates. The last sequence number in a most recent datagram transmitted from base station for every connection is also stored and is used to determine the location where the datagram was lost or corrupted when destination node is failed to successfully receive the segment. Before retransmission of any packet from the base station, the checksum is calculated to find out the correctness of the datagram.

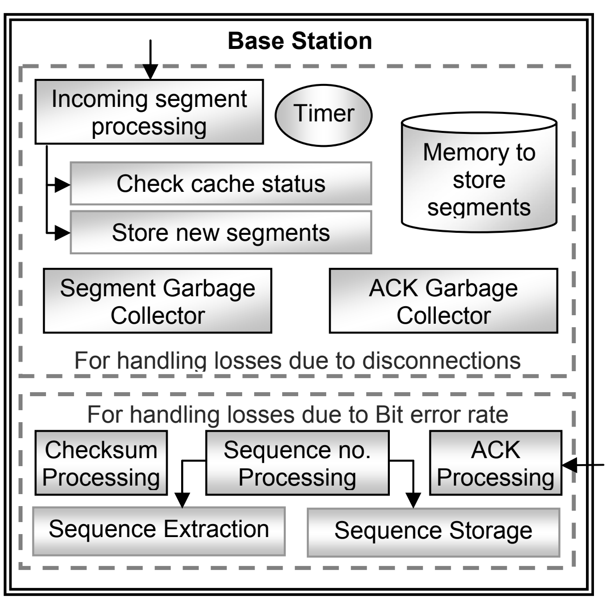

Figure 1: TCP-DR Architecture

Therefore, the disconnections or mobility is determined by timeout at the base station while corruption of segment is determined by comparing the sequence numbers from the acknowledgement packets. For the implementation of TCP-DR, the following architecture is proposed as shown in figure 1.

Incoming segment processing: This module determines that the received datagram is either a retransmitted or a new segment. It is done by comparing stored sequence number at the base station with last sequence number in an incoming datagram. In both case, new datagram is stored or replaced appropriately as shown in flow chart in figure 2.

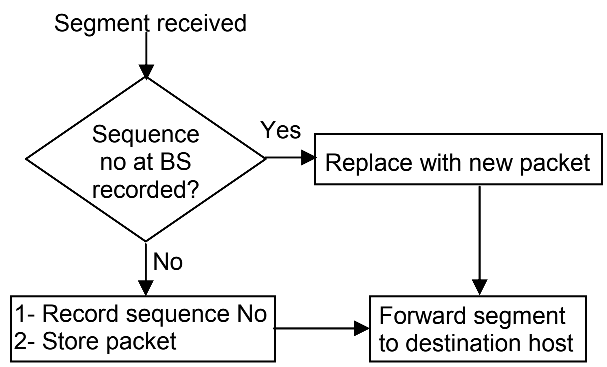

Figure 2: Data processing at base station

Memory to store packets: There is a memory allocated to store the packets coming from the sending host. These packets are temporarily stored until an ACK from the destination host is received.

Segment garbage collector: This module removes those datagram that are successfully delivered to the destination node. Each incoming datagram are stored and retained until their acknowledgement is received.

ACK garbage collector: This module purge unnecessary stored ACKs and their sequence numbers at base station.

Timer: There is a timestamp allocated to each packet passing through base station and local time out occurs when an ACK of that packet is not received. Its ultimate aim is to determine the connectivity of the receiving host.

Checksum processing: When destination host detects that the received packet is corrupted, it generates an ACK with last successfully received packet sequence number. On repetition of this ACK, base station applies checksum to locally stored packet with the same sequence number. The ultimate target is to retransmit a packet locally from base station instead of retransmission from data sending host. So that unnecessary traffic from the network is removed and TCP connection is recovered in a minimal time interval.

Sequence no processing: When source host sends a packet, the base station at intermediate location has a module that extracts and stores the sequence number from the packet. The sequence number is used to detect the location of loss of packet (either wired or wireless).

ACK processing: This module compares the incoming ACK sequence number with the stored sequence number. Greater sequence number verifies the successful delivery of data packet otherwise checksum module performs its function to determine the correctness of stored segment. If stored packet is corrupted than this module generates ACK_{C-CLN} to inform the source host that packet is corrupted in wired or wireless link as show in figure 3 by setting / resetting Ob and Db bits in the reserved portion of normal TCP packet.

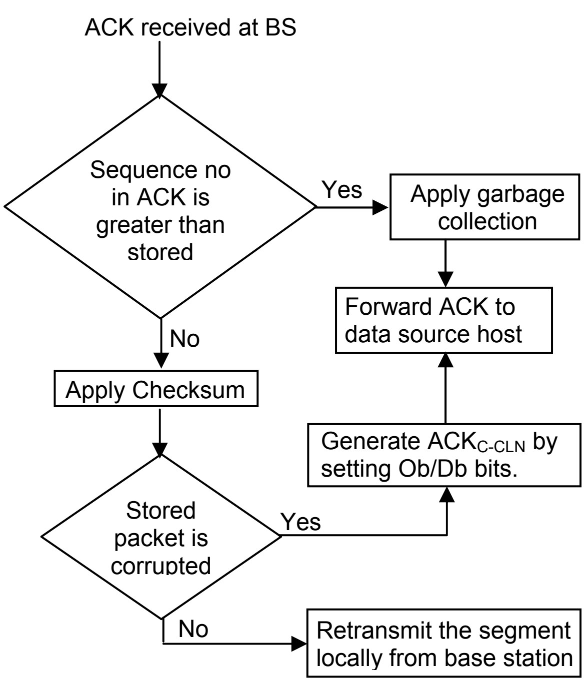

Figure 3: ACK processing at Base Station

5.2 TCP-DR at data sending host

There are three possible actions by the TCP at the data sending host. First one is that source node can initiate congestion control mechanism when congestion occurs at any link on the connection path as in normal TCP connections. Secondly, source node can initiate fast retransmission when segments are corrupted or lost and third possibility can pause its transmission when mobile node is disconnected from the base station instead of terminating TCP sessions. The architecture of TCP-DR at data sending host is shown in figure 4.

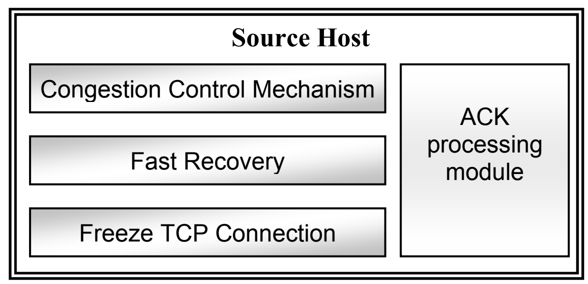

Figure 4: TCP-DR at data sending host

The data transmitting node selects one option out of these three possibilities. Congestion Control mechanism is initiated when an ACK packet is received three times whose sequence number is smaller than last transmitted sequence number, window size is greater than zero and Ob and Db bits are zero. This kind of acknowledgement shows that there is congestion in the network. Fast recovery is initiated when the packet is lost or corrupted in wireless link that is determined by Ob and Db bits in ACK_{C-CLN} segment. In the case of source host is mobile nodes, Ob bit set show that packet is lost in wired link and vice versa. And TCP connection is being frozen, when mobile node is disconnected from the base station. Disconnections are determined by the timer at base station

6.0 Conclusion

Physical links are transparent from TCP, therefore packet losses due to high bit error rate in wireless links or mobile node disconnections are falsely considered as due to congestion by TCP. It results in reducing transmission speed by initiating congestion control mechanisms. As a result already lesser available bandwidth for wireless links is wasted. TCP either should be capable of distinguishing various reasons of packet losses or such non-congestion-related losses should be reduced. In this paper, we carried out a brief survey on the challenges; the TCP has come across, in wired-cum-wireless networks and analyzed different protocols developed to improve performance in wireless networks and designed TCP-DR scheme that inherits the positive aspects of M-TCP and E-TCP. M-TCP is best protocol to handle the mobility and handoff whereas E-TCP is extremely efficient to handle packet losses due to bit error rate. Our proposed hybrid scheme works fine for the optimal utilization of available bandwidth for wireless links.

In future, we will simulate proposed scheme in NS-2 and standardize TCP-DR at transport layer in such a way that it incorporates other performance factors discussed in section 2 and also concentrate on how to handle TCP-DR, when $ACK_{C\text{-}CLN}$ is lost.

7.0 References

[1] Ahmad, Ejaz and Waqar, "Transport layer Performance: Improvement in Infrastructure based Wireless Networks", HONET'04, Pakistan, Dec 2004.

[2] Fonseca, Nahur; Crovella, Mark, "Bayesian Packet Loss Detection for TCP", BUCS Technical Report, USA, July 1, 2004.

[3] Bakre and B.R. Badrinath, "I-TCP: Indirect TCP for mobile hosts," in proc. of 15th International. Conference on Distributed Computing System (ICDCS), May 1995.

[4] B. Hari, S. Srinivasan, A. Elan, R. H. Katz, "Improving TCP/IP Performance over Wireless Networks", In Proc. 1st ACM Int'l Conf. on Mobile Computing and Networking (Mobicom), Nov 1995.

[5] R. Braden, "Requirements for Internet Hosts Communication Layers", RFC1122, Oct 1989.

[6] Bakshi, B., Krishna, P. et al., "Improving performance of TCP over wireless networks" 17th International Conference on Distributed Computing Systems, 1997, pp. 365–373.

[7] A. Bakre and B. R. Badrinath, "I-TCP: Indirect TCP for Mobile Hosts", Technical Report DCS-TR-314, Rutgers University, October 1994.

[8] W. G. Zeng, M. Zhan, Z. Li, and Lj. Trajkovic, "Improving TCP Performance with Periodic Disconnections over Wireless Links", *OPNETWORK 2003*, Washington, DC, Aug. 2003.

[9] Hala Elaarag, "Improving TCP performance over mobile networks", ACM Computing Surveys, Vol. 34, No. 3, September 2002, pp. 357–374.

[10]M. Mathis, J. Mahdavi, S. Floyd, A. Romanow, "TCP Selective Acknowledgement Options", RFC2018, October 1996

[11]K. Dzmitry, G. Fabrizio, "A Cross-layer scheme for TCP Performance Improvement in Wireless LANs", Publications Technical Report, DIT, February 2004.

[12] Chandra D, Harris R J, Shenoy N, "TCP Performance for Future IP-Based Wireless Networks", 3rd IASTED International Conference on Wireless and Optical Communications (WOC2003), Banff Canada, July 2003.

[13] Yavatkar R, Bhagawat N, "Improving end-to-end performance of TCP over mobile internetworks", Proceedings of the Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, 1995, pp. 146–152.